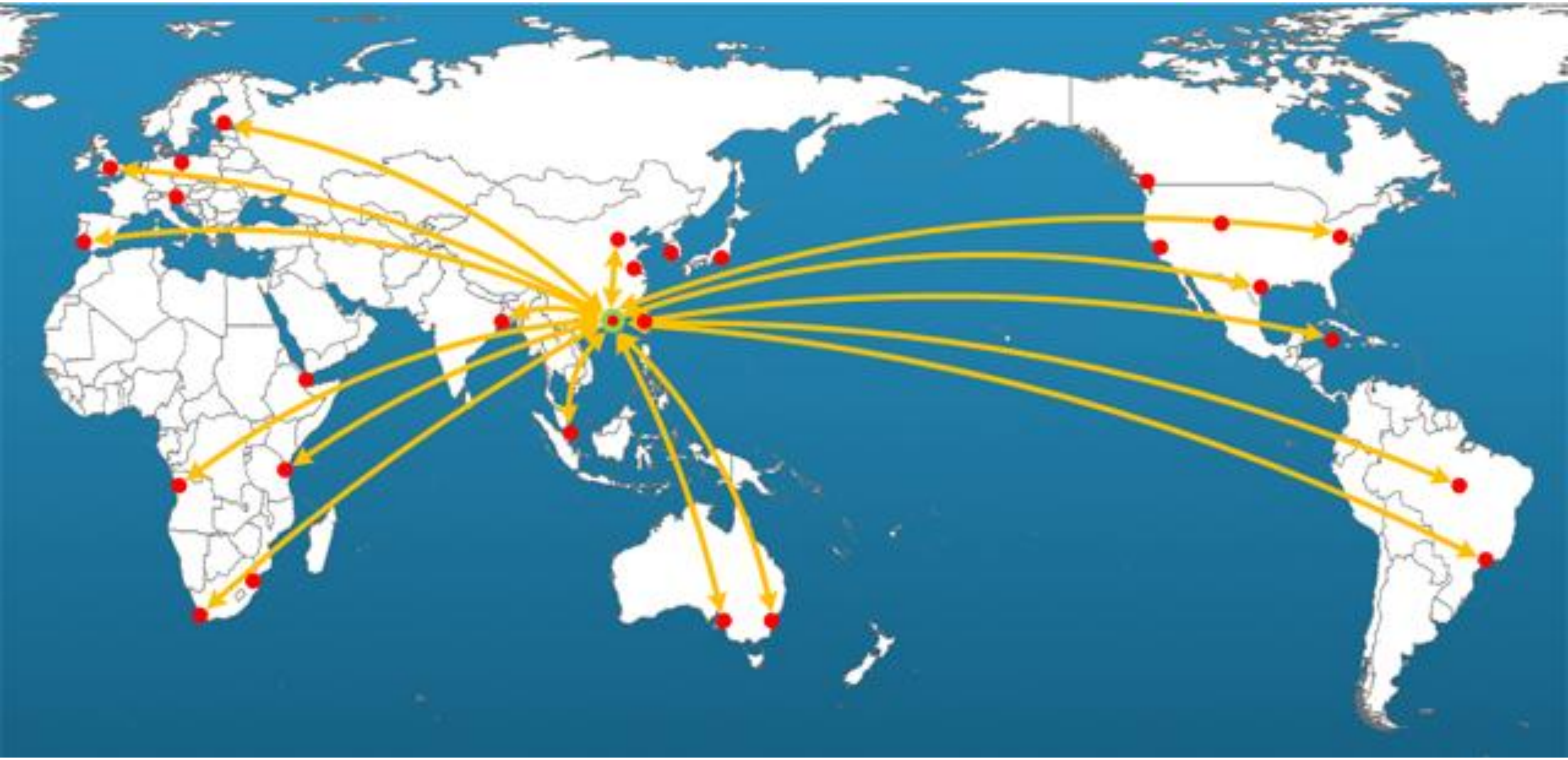
Active Measurement of Data-Path Quality in a Non-cooperative Internet

Rocky K. C. Chang
Internet Infrastructure and Security Group

7 Oct. 2010

Can we measure Internet path from any point to any point today?



Can we measure Internet path from any point to any point today?

- Not quite
- Proxies and middleboxes
- Many require another protocol from both endpoints (e.g., OWAMP, TWAMP)
- Non-cooperative
  - Using/hacking existing protocols
  - ping, tulip, pathneck …

Can we measure Internet path from any point to a large number of points?

- Could be
- Some nodes = end hosts
- Sting for loss; Sprobe and DSLprobe for capacity; Dual connection test for reordering …

(Invalid) assumptions

- Control-path quality = data-path quality
  - ICMP, TCP SYN, TCP RST
- Middleboxes not an issue
  - Dropping, rate-limiting, additional latency
- No changes in systems
  - Consecutive increment of IPID (e.g., tulip)
- Sampling rate not an issue

(Invalid) assumptions

- Control-path quality = data-path quality
  - ICMP, TCP SYN, TCP RST
- Middleboxes not an issue
  - Dropping, rate-limiting, additional latency
- No changes in systems
  - Consecutive increment of IPID (e.g., tulip)
- Sampling rate not an issue

Invalid assumptions beget unreliable measurement.

Other problems in practice

- Support only one or two metrics
- Round-trip measurement
- No control over packet sizes
- Not integrated with application protocols

Other problems in practice

- Support only one or two metrics
- Round-trip measurement
- No control over packet sizes
- Not integrated with application protocols

Practical issues stifle deployment.

Our design principles

- Use normal data packet to measure data-path quality.
- Use normal and basic data transmission mechanisms
- Integrated into normal application sessions.

Our design principles

- Use normal TCP data packet to measure data-path quality.
- Use normal and basic TCP data transmission mechanisms specified in RFC 793.
- Integrated into normal HTTP application sessions.

Our design principles

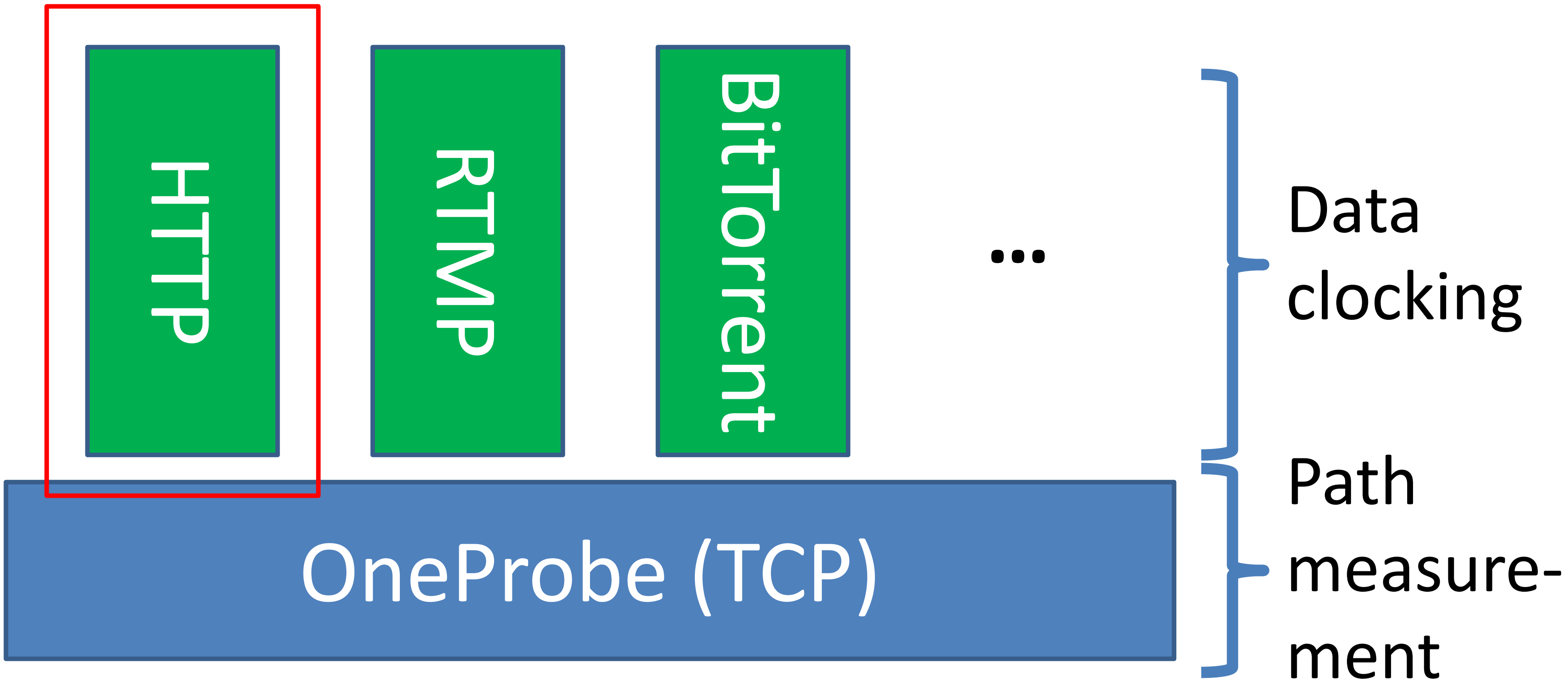
- Use normal TCP data packet to measure data-path quality.
- Use normal and basic TCP data transmission mechanisms specified in RFC 793.
- Integrated into normal HTTP application sessions.

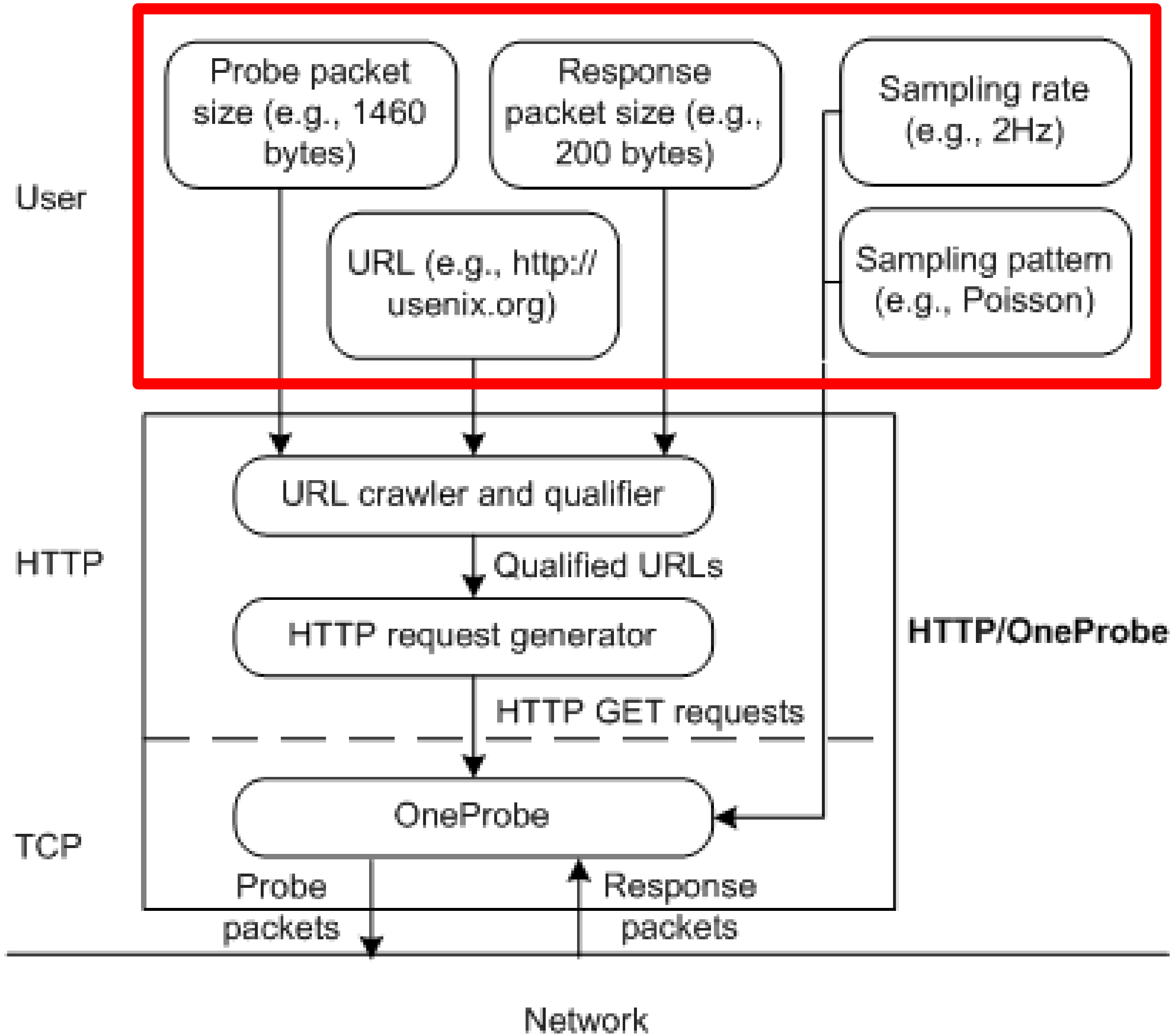
Reliable measurement

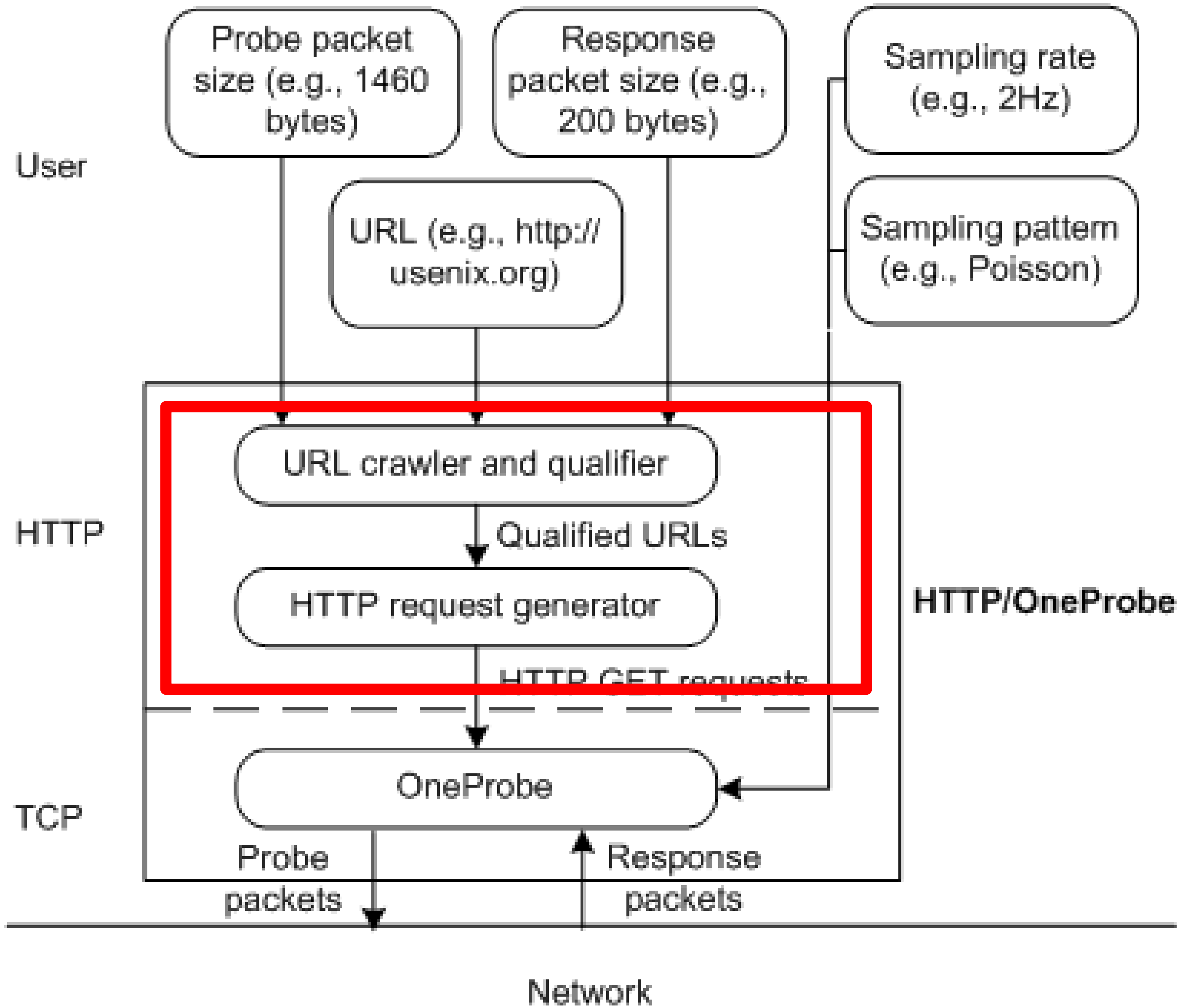
What does HTTP/OneProbe offer?

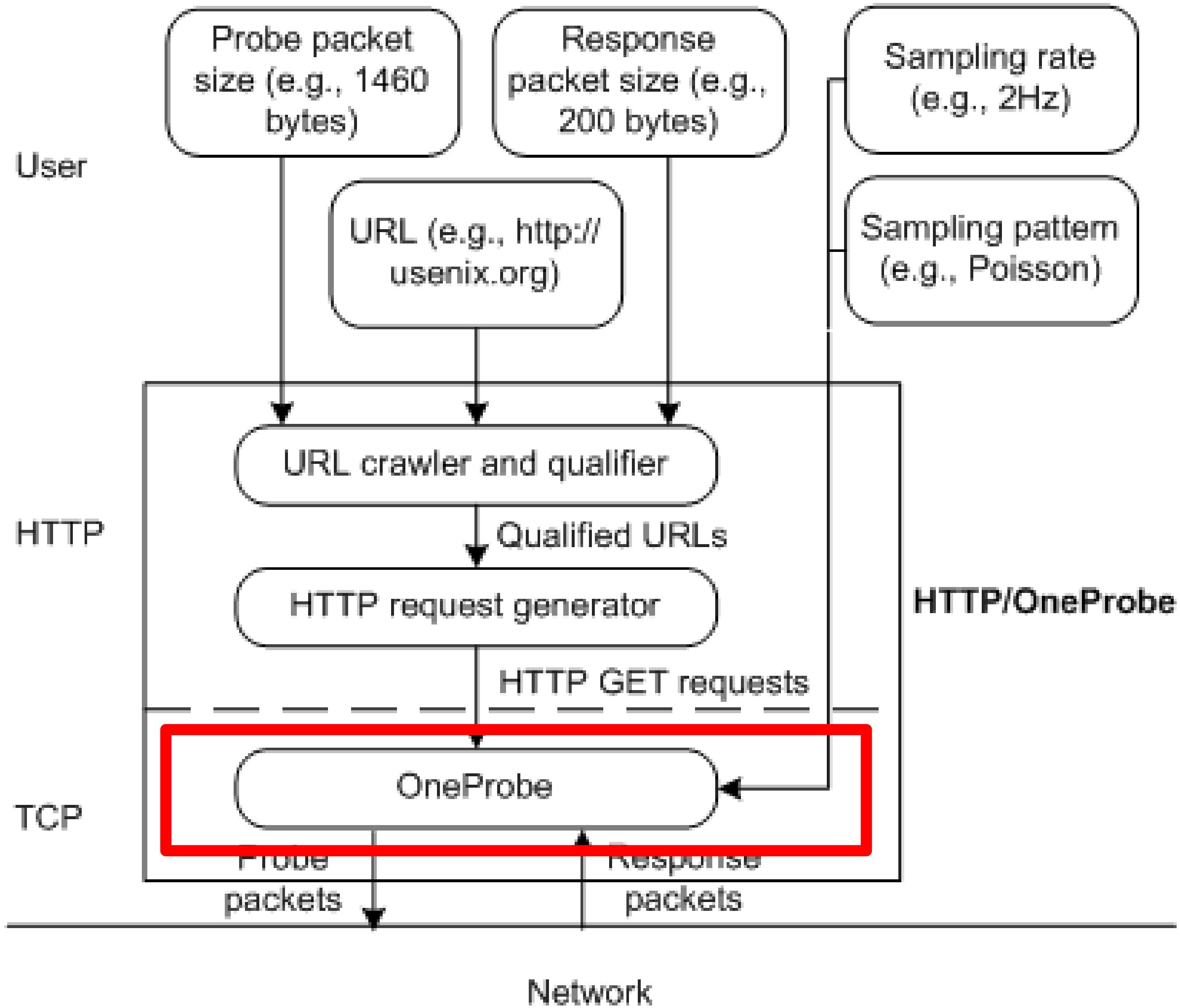
- Continuous path monitoring in an HTTP session (stateful measurement)
- All in one:
  - Round-trip time
  - Loss rate (uni-directional)
  - Reordering rate (uni-directional)
  - Capacity (uni-directional)
  - Loss-pair analysis
  - …

HTTP/OneProbe




Probe packet size (e.g., 1460 bytes)
Response packet size (e.g., 200 bytes)
Sampling rate (e.g., 2Hz)
User
URL (e.g., http:// usenix.org)
Sampling pattern (e.g., Poisson)
URL crawler and qualifier
HTTP
Qualified URLs
HTTP request generator
HTTP/OneProbe
HTTP GET requests
OneProbe
TCP
Probe packets
Response packets
Network


Probe packet size (e.g., 1460 bytes)
Response packet size (e.g., 200 bytes)
Sampling rate (e.g., 2Hz)
User
URL (e.g., http:// usenix.org)
Sampling pattern (e.g., Poisson)
URL crawler and qualifier
HTTP
Qualified URLs
HTTP request generator
HTTP/OneProbe
HTTP GET requests
OneProbe
TCP
Probe packets
Response packets
Network


Probe packet size (e.g., 1460 bytes)
Response packet size (e.g., 200 bytes)
Sampling rate (e.g., 2Hz)
User
URL (e.g., http://usenix.org)
Sampling pattern (e.g., Poisson)
URL crawler and qualifier
HTTP
Qualified URLs
HTTP request generator
HTTP/OneProbe
HTTP GET requests
OneProbe
TCP
Probe packets
Response packets
Network

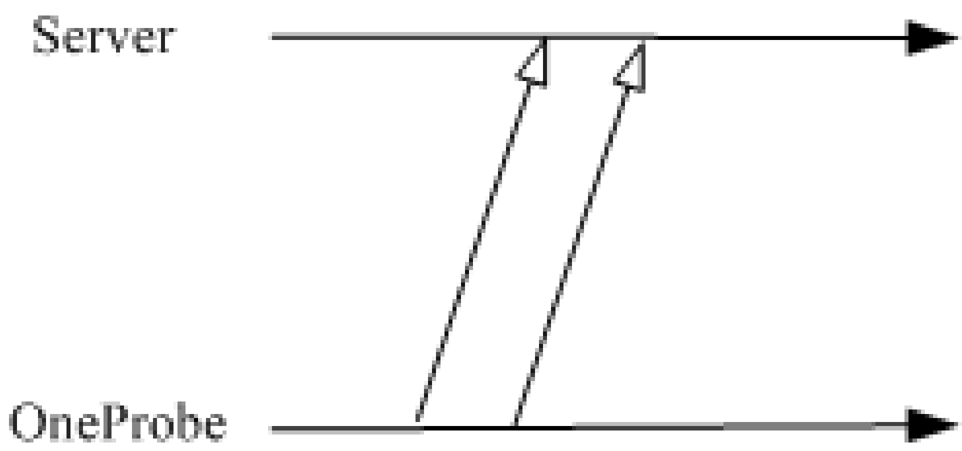
The remaining talk

- OneProbe
- HTTP/OneProbe
- Validation
- Measurement results
- Conclusions and current works

OneProbe

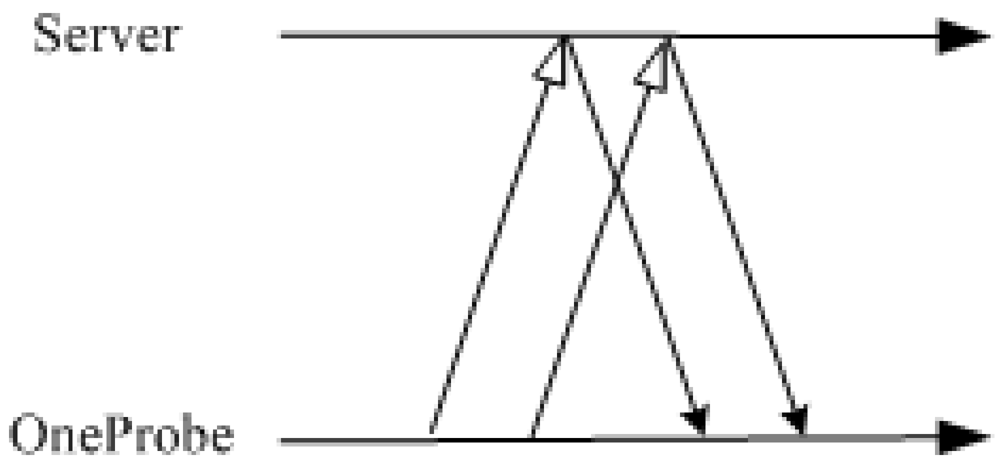
The probe design

- Send two back-to-back probe data packets.
  - Capacity measurement based on packet-pair dispersion
  - At least two packets for packet reordering
  - Determine which packet is lost.



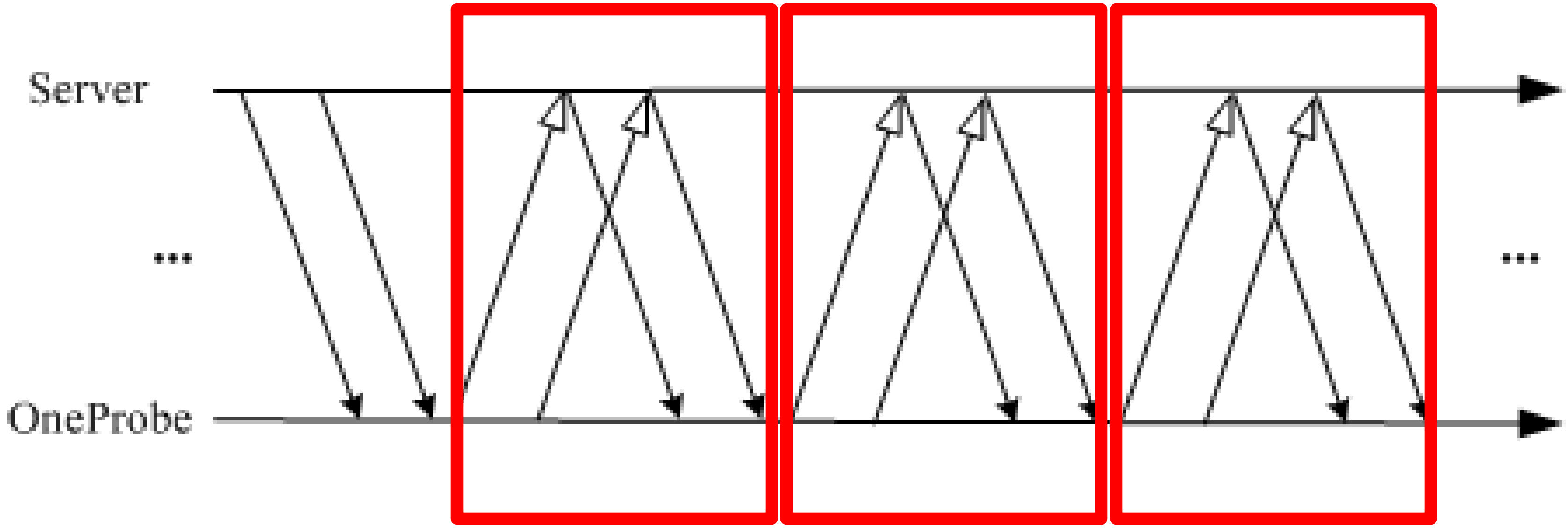
The probe design (cont'd)

- Similarly for the response packets

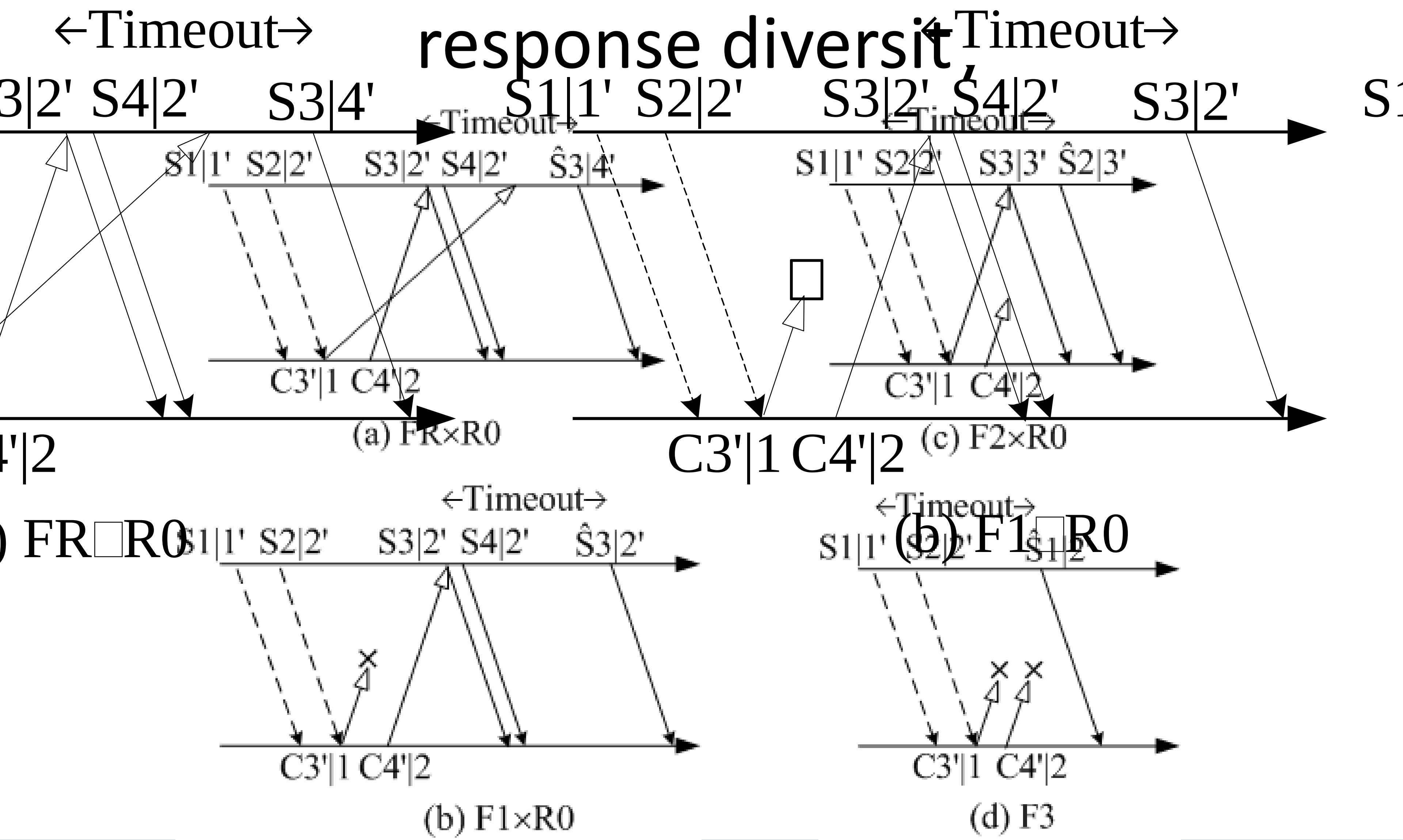


- Each probe packet elicits a response packet.
  – Adv. Window = 2 and acknowledge only 1 packet.

Bootstrapping and continuous monitoring



Loss and reordering measurement via response diversity



18 possible path events

	R0	RR	R1	R2	R3
F0	√	√	√	√	√
FR	√	√	√	√	√
F1	√	√	√	√	√
F2	√	–	√	–	–
F3	–	–	–	–	–

and their response packets

Path events	1st response packets	2nd response packets	3rd response packets
1. F0 $\times$ R0	$S3\|3'$	$S4\|4'$	–
2. F0 $\times$ RR	$S4\|4'$	$S3\|3'$	–
3. F0 $\times$ R1	$S4\|4'$	$\widehat{S}3\|4'$	–
4. F0 $\times$ R2	$S3\|3'$	$\widehat{S}3\|4'$	–
5. F0 $\times$ R3	$\widehat{S}3\|4'$	–	–
6. FR $\times$ R0	$S3\|2'$	$S4\|2'$	$\widehat{S}3\|4'$
7. FR $\times$ RR	$S4\|2'$	$S3\|2'$	$\widehat{S}3\|4'$
8. FR $\times$ R1	$S4\|2'$	$\widehat{S}3\|4'$	–
9. FR $\times$ R2	$S3\|2'$	$\widehat{S}3\|4'$	–
10. FR $\times$ R3	$\widehat{S}3\|4'$	–	–
11. F1 $\times$ R0	$S3\|2'$	$S4\|2'$	$\widehat{S}3\|2'$
12. F1 $\times$ RR	$S4\|2'$	$S3\|2'$	$\widehat{S}3\|2'$
13. F1 $\times$ R1	$S4\|2'$	$\widehat{S}3\|2'$	–
14. F1 $\times$ R2	$S3\|2'$	$\widehat{S}3\|2'$	–
15. F1 $\times$ R3	$\widehat{S}3\|2'$	–	–
16. F2 $\times$ R0	$S3\|3'$	$\widehat{S}2\|3'$	–
17. F2 $\times$ R1	$\widehat{S}2\|3'$	–	–
18. F3	$\widehat{S}1\|2'$	–	–

Path event distinguishability

- All 18 cases can be distinguished except for
  - A1. F1×R2 and F1×R3
  - A2. F1×RR and F1×R1
  - A3. F0×R3 and FR×R3
- Resolving the ambiguities
  - A1 and A2: use RTT.
  - A3: use TCP timestamping.

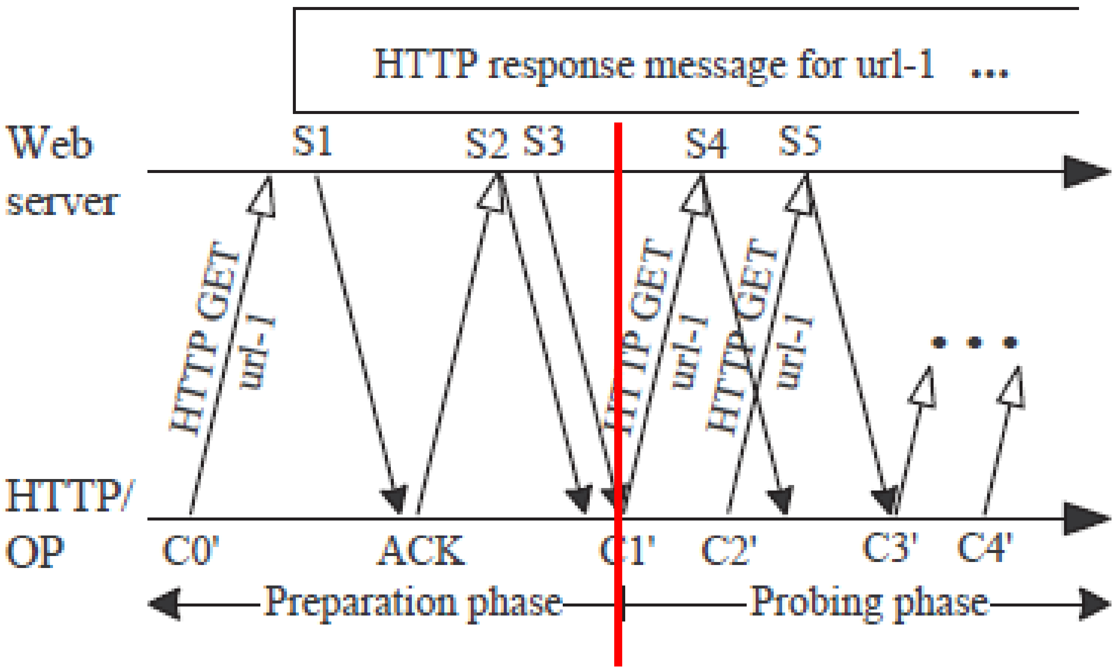
Other issues

- Use TCP ACKs to improve performance
- When to start a new probe round when timeout is involved
- Use concurrent TCP connections to increase sampling rate.

HTTP/OneProbe

Preparation and probing phases

- Ramp up the server's cwnd to 2.
- Write the requested object into the kernel.



Finding qualified http URLs

- A qualified http URL:
  - its HTTP GET request can be retrofitted into a probe packet, and
  - the GET request can induce at least 5 response packets from the server.
- Verify whether a user-specified URL meets the size requirement for the response packets.
- The HTTP GET request for a qualified URL must induce a 200(OK) response.

Preparing the HTTP GET requests

- To craft a probe packet for an HTTP request, expand the packet size through the Referer field and a customized string.
- Exploit the HTTP/1.1's request pipelining to include a GET message in each probe packet.

Validation

Web server software and operating systems

- 100% passed

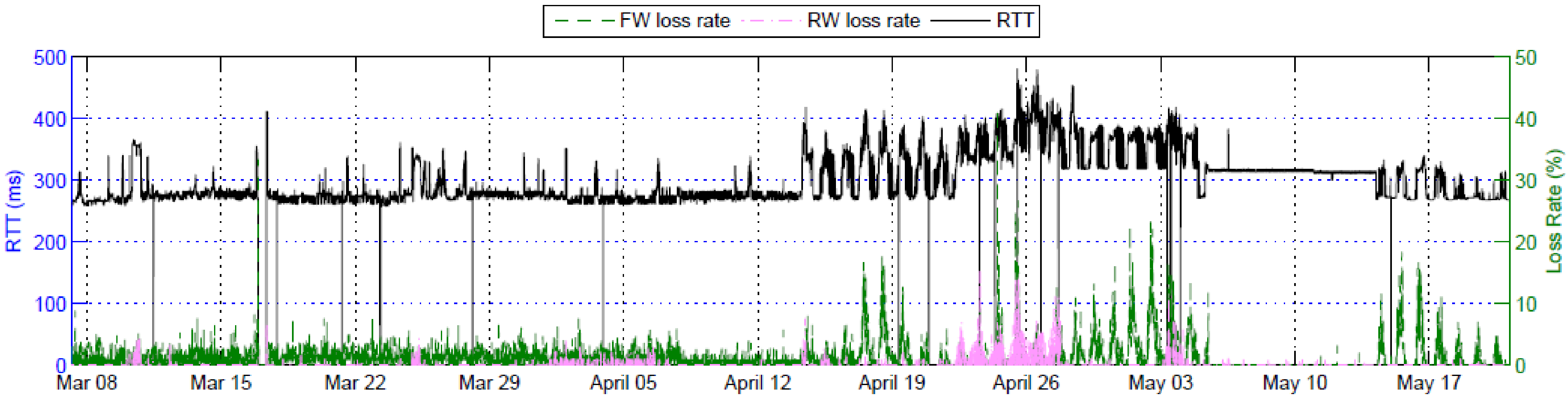
Systems tested in our lab.:	FreeBSD v4.5/4.11/5.5/6.0/6.2, Linux kernel v2.4.20/2.6.5/2.6.11/2.6.15/2.6.18/2.6.20, MacOSX 10.4 server, NetBSD 3.1, OpenBSD 4.1, Solaris 10.1, Windows 2000/XP/Vista
Systems tested in the Internet:	AIX, AS/400, BSD/OS, Compaq Tru64, F5 Big-IP, HP-UX, IRIX, MacOS, NetApp NetCache, NetWare, OpenVMS, OS/2, SCO Unix, Solaris 8/9, SunOS 4, VM, Microsoft Windows NT4/98/Server 2003/2008
Servers tested in our lab.:	Abyss, Apache, Lighttpd, Microsoft IIS, Nginx
Servers tested in the Internet:	AOLserver, Araneida, Apache Tomcat, GFE, GWS-GRFE, IBM HTTP Server, Jetty, Jigsaw, LiteSpeed, Lotus-Domino, Mongrel, Netscape-Enterprise, OmniSecure, Oracle HTTP Server, Orion, Red Hat Secure, Redfoot, Roxen, Slinger, Stronghold, Sun Java System, thttpd, Twisted Web, Virtuoso, WebLogic, WebSiphon, Yaws, Zeus, Zope

Web servers in the wild

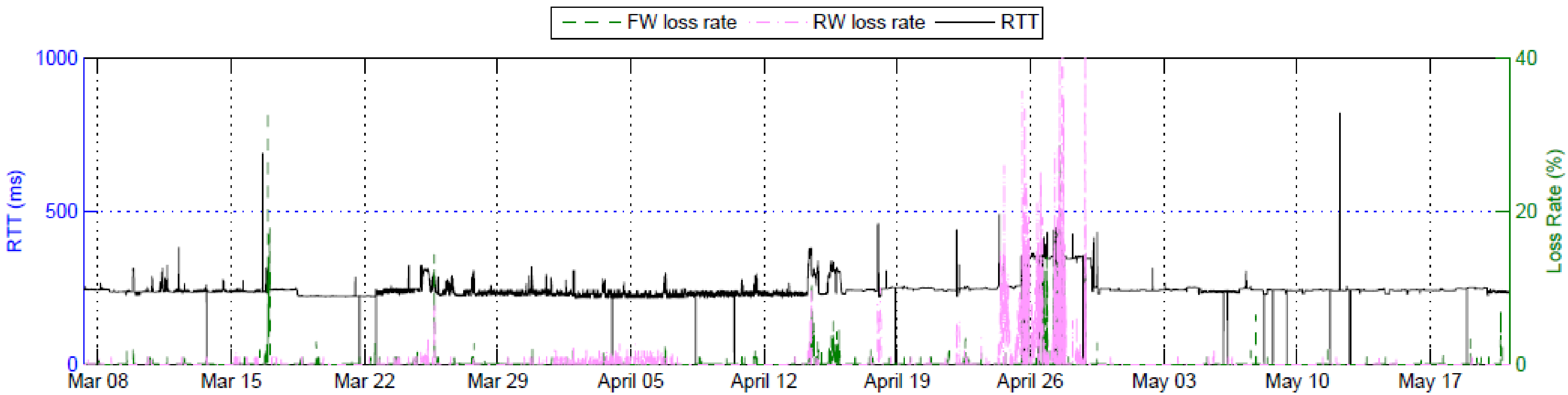
- Tested 37,874 websites randomly selected.
- Successful (93.00%): These servers passed all tests.
- Failures in the preparation phase (1.03%): OneProbe could not start the probing phase.
- Failures in testing in-ordered probes (0.26%)
- Failures in testing out-ordered probes (5.71%)

Measurement results

Time series of RTT and loss
HK → Europe and UK
April 2010



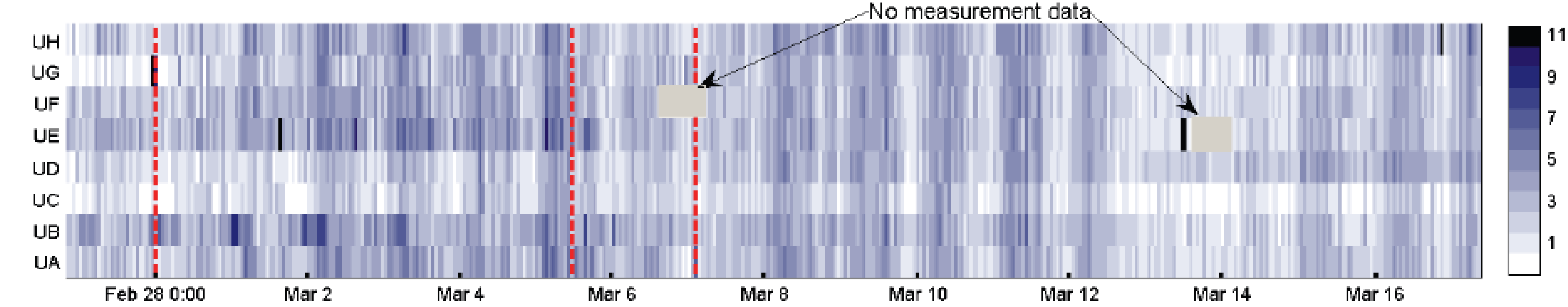
(a) UB→NOK's RTT and loss rates.



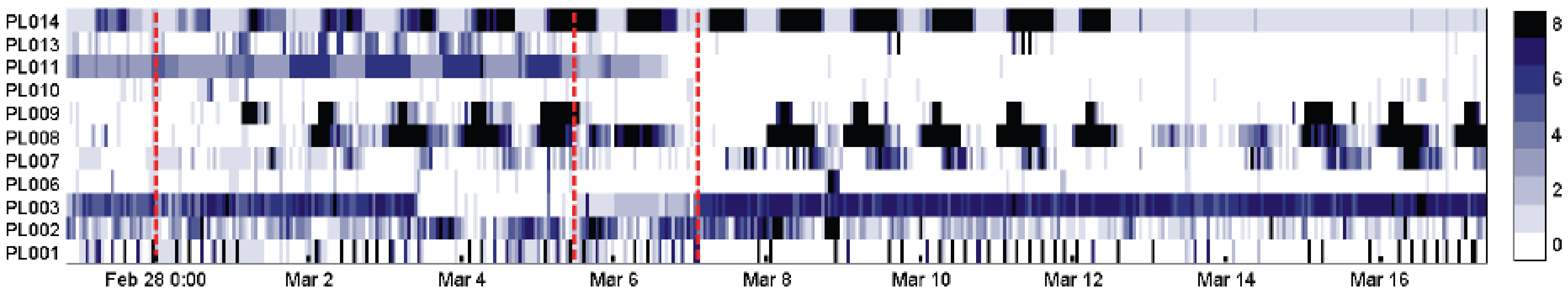
(b) UB→BBC's RTT and loss rates.

Heat maps for packet loss
8 sources at HK → 11 PlanetLab nodes
Feb-Mar 2010

Forward paths

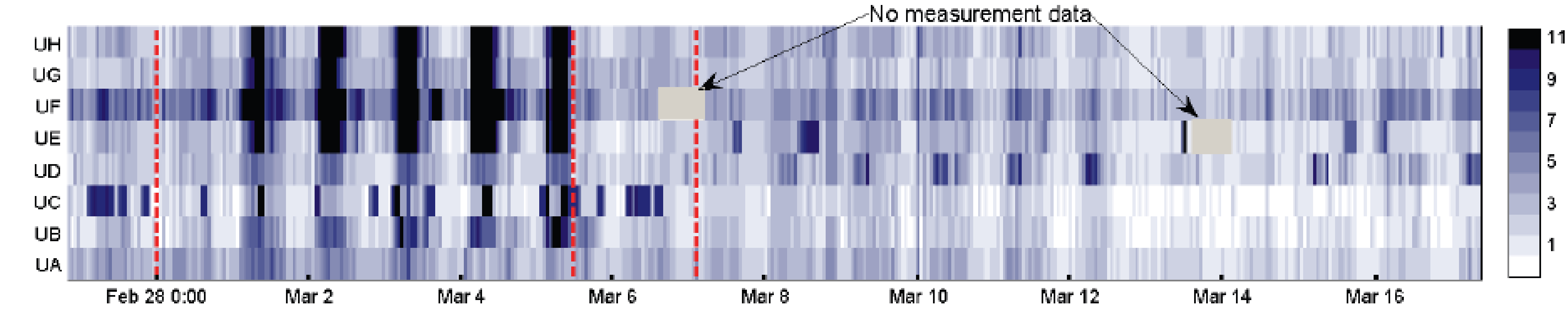


(a) Forward paths (grouped by sources).

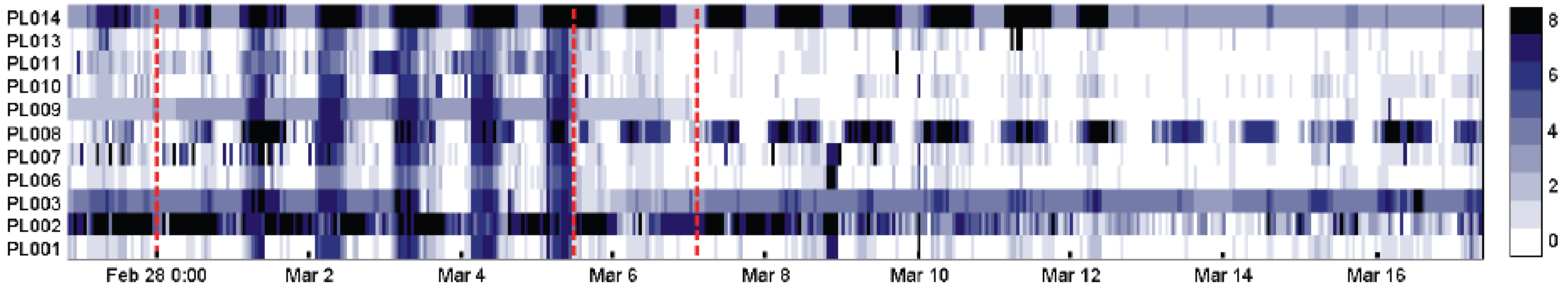


(b) Forward paths (grouped by destinations).

Reverse paths

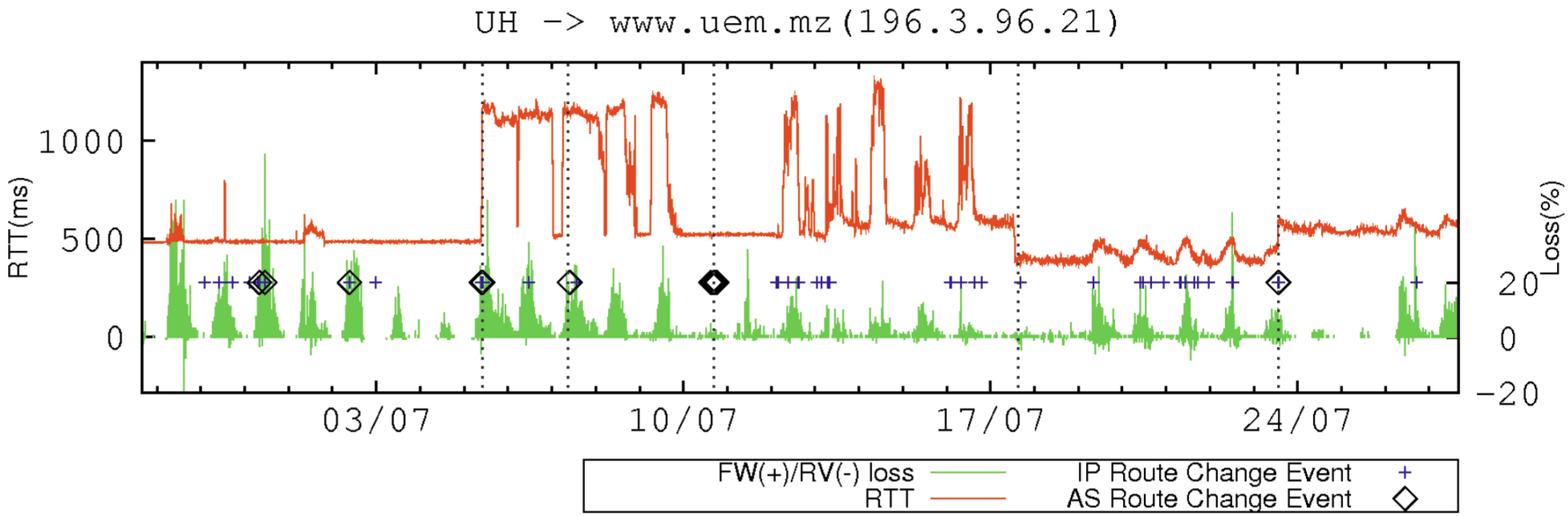


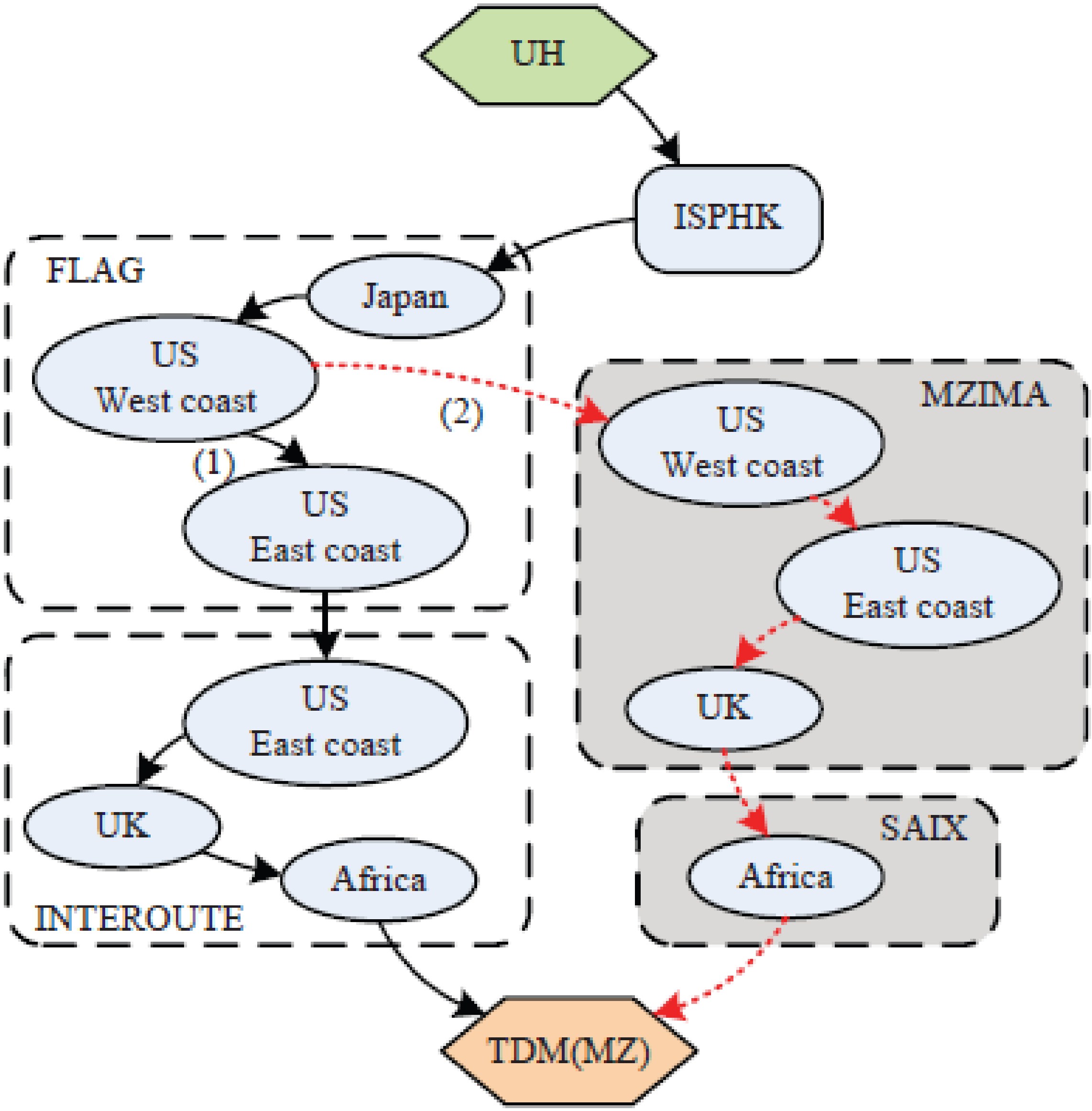
(c) Reverse paths (grouped by sources).



(d) Reverse paths (grouped by destinations).

Time series of RTT and loss + route changes HK → Mozambique July 2010


UH -> www.uem.mz(196.3.96.21)
RTT(ms)
1000
500
0
Loss(%)
20
0
−20
03/07
10/07
17/07
24/07
FW(+)/RV(-) loss
RTT
IP Route Change Event
AS Route Change Event


UH
ISPHK
FLAG
Japan
US
West coast
(2)
(1)
US
East coast
MZIMA
US
West coast
US
East coast
UK
US
East coast
UK
Africa
INTEROUTE
SAIX
Africa
TDM(MZ)

Other TCP-based methods

- Sting by Salvage: loss measurement
  - Reverse measurement by TCP ACKs
- A suite of methods for reordering measurement by Bellardo and Savage
  - TCP ACKs, TCP SYN-RST
- Sprobe by Saroiu, Gummadi and Gribble
  - TCP RSTs

Conclusions and current works

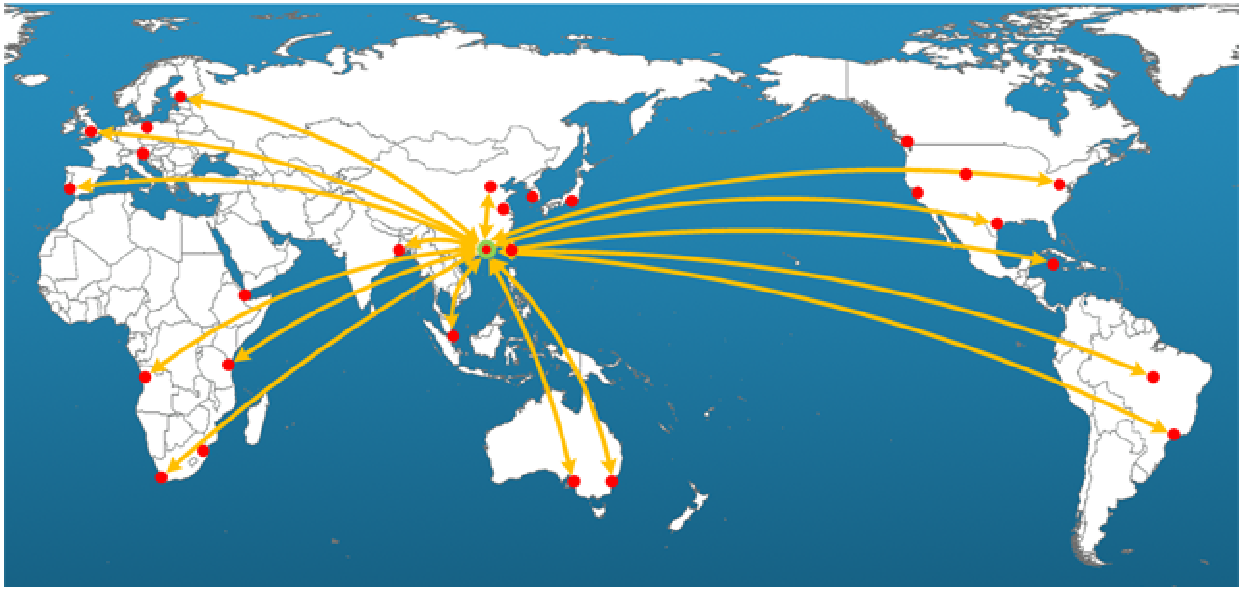
- Turn a network protocol into a measurement protocol.
- Coming up a novel measurement method is just half a story.
- Making it work in the non-cooperative Internet is hard.
- Set the goal for a real-world deployment is crucial.
- Current works
  – Enriching OneProbe's capability
  – Applications: e.g., network tomography

Contact us

OneProbe

Measuring Non-cooperative Internet Paths for Quality

Home | Publications | Projects | Dataset and Tools | Team



Latest News

[23 September 2010] Edmond defended his PhD thesis successfully.

[19 July 2010] Our paper on "Measurement of Loss Pairs in Network Paths" is accepted in ACM/USENIX IMC 2010.

[23 June 2010] Our paper on "Could Ash Cloud or Deep-Sea Current Overwhelm the Internet?" is accepted as a poster in HotDep.

[30 December 2009] Received an ITSP funding on "Reliable and Accurate Bandwidth Measurement of Asymmetric Network Paths" and another year of funding for the HARNET measurement project.

OneProbe is a new method for measuring a non-cooperative path's quality. The measurement is conducted in one or more concurrent TCP connections by an endpoint of the path under measurement. The main novelty is the capability of measuring, in addition to round-trip time, unidirectional packet loss rate, packet reordering rate and capacity from the same packet-pair probes. Other advantages include

- Traversing most of the middleboxes placed before web servers in the Internet,
- Measuring the path quality experienced by legitimate data packets (not control packets),
- Allowing users to configure the packet size for the probe and response packets, and
- Allowing users to configure a high sampling rate and different sampling patterns.

More details here.

More information

- “Measurement of Loss Pairs in Network Paths,” Proc. ACM/USENIX IMC, November 2010.
- "Could Ash Cloud or Deep-Sea Current Overwhelm the Internet?" Proc. USENIX HotDep, October 2010 (a poster).
- “A Minimum-Delay-Difference Method for Mitigating Cross-Traffic Impact on Capacity Measurement,” Proc. ACM CoNext, December 2009.
- “Design and Implementation of TCP Data Probes for Reliable Network Path Monitoring,” Proc. USENIX Annual Tech. Conf., June 2009.

Thanks